Clustering

Lesson 3 : Lab Session

Advanced Machine Learning, CentraleSupelec

Teacher's Assistant: Omar CHEHAB

Professors : Emilie CHOUZENOUX, Frederic PASCAL

General Information

- **Assignment :** alone or in pairs, you will code the algorithms you learnt in 'scikit-learn formalism', and apply them to images and text.

- **Due :** the 5 lab assignments for lessons 3-7 are due a week from when they are given, at aml.centralesupelec.2020@gmail.com

- **Grading** : each assignment is worth 4 points — your 4 best labs out of the 5 will be retained and will count for half of your final grade.

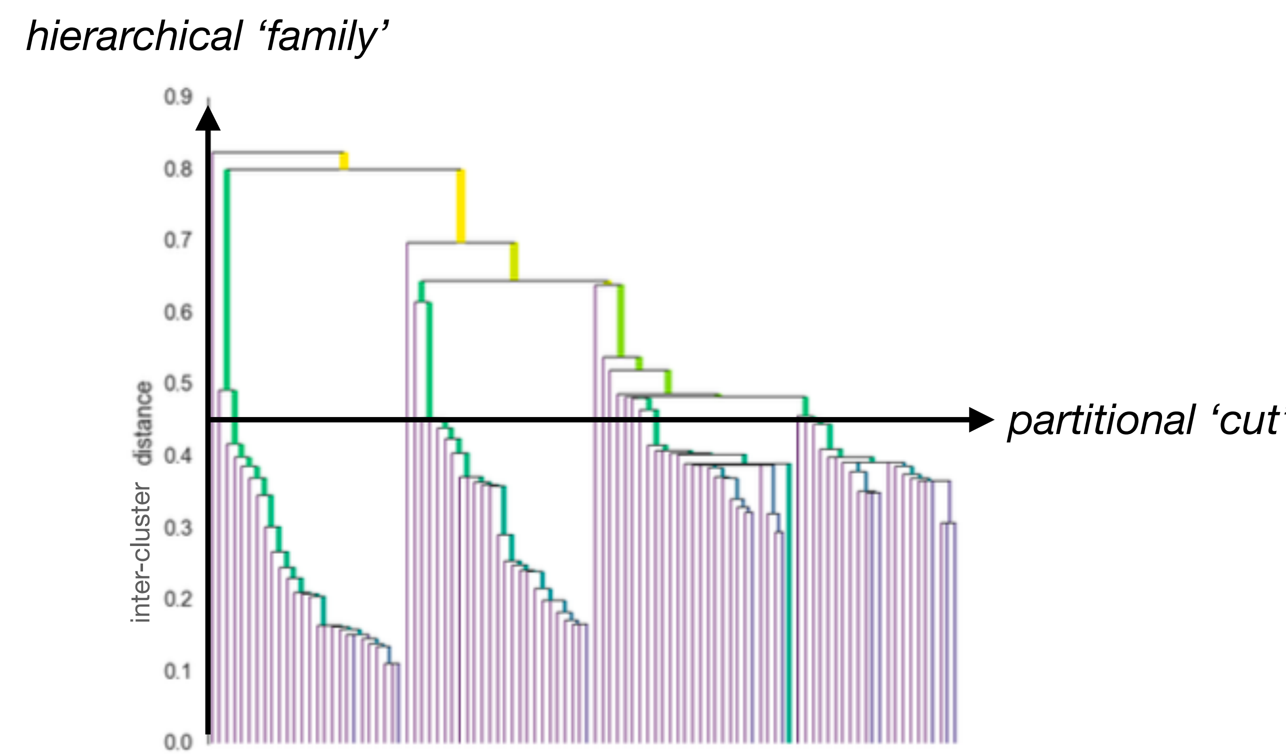
- **Questions :** questions or feedback are welcome after class or by email at l-emir-omar.chehab@inria.fr

Lesson: recap

	type	n_clusters	Objective	Algorithm	Robust to	Clusters
K-Means	partitional	hardcoded	$\min_{\delta_{ik}, c_k} \delta_{ik} \sum_{k=1}^{K} \sum_{i=1}^{m} \left\| x^i - c_k \right\|^2$ cluster sets (location and assign.) within-cluster variance	alternatively assign points to clusters, recompute clusters as center-of-points		Points that are <u>near..</u>
Agglomerative Single-Linkage	hierarchical (bottom-up: merge)	given by… ‘cutoff’ ε	-	sequentially compute distance (e.g. min) between clusters and merge the two nearest clusters, until you end up with one cluster.	init	 …<u>nearest</u>
DBSCAN	partitional	given by… ‘cutoff’ ε density minPts	-	Identify core points as having at least minPts in their ε-neighborhood. Their connected components on the ε-neighbor graph make the clusters. Non-core points either join an ε-nearby cluster, else are noise.	…and outliers, noise	… and in <u>dense</u> regions
HDBSCAN	hierarchical (top-down: split)	given by… ~~‘cutoff’ ε~~ density minPts	-	1. Build complete graph weighted by specific metric that penalizes sparsity* 2. Extract the minimum spanning tree 3. Construct a cluster hierarchy of connected components by removing heaviest edges 4. Condense the cluster hierarchy based on a min. cluster size before merge (less is noise) 5. Extract the clusters with long antecedance (robust to cutoff) in the condensed tree : tunes ε for each cluster.	…and n_clusters	… that are <u>not easily split</u>

*for two ‘close’ points,
clamp their distance to that to the farthest Minpts neighbor.

From a modelling standpoint



A *partitional* clustering can sometimes be framed as the ‘cutoff’ of a *hierarchical* clustering, i.e. as the *instance* of a *relaxed* problem in which it is embedded.

For e.g., DBSCAN (**partitional**) can be understood as the ε-‘cut’ of HDBSCAN (**hierarchical, top-down**) without steps 4 and 5, or of Agglomerative Single-Linkage (**hierarchical, bottom-up**) where the space is transformed s.t. sparse points (‘not having a core-point eps-neighbor’) are farther away*.

* transforming thusly the space is equivalent to keeping the original space but modifying the metric to that of Step 1 of HDBSCAN

Assignment: plan

1. K-Means (*scikit-learn*)

2. Agglomerative Single-Linkage (*your own code*)

3. DBSCAN (*scikit-learn*)

4. HDBSCAN (*scikit-learn*)

5. Applications : clustering observations on Mars and color-reduction (*scikit-learn*)